REMARKS

Applicant respectfully requests the reconsideration of this application and the consideration of the following remarks.

The information disclosure statement filed 12/31/01 was objected to for not including a publication date for the reference "Proposed SMPTE Standard for Television, SMPTE314M". According to the web site of SMPTE (e.g., http://www.smpte.org/smpte_store/standards/index.cfm?scope=1&CurrentPage=13&stdtype=smpte), SMPTE314M was issued in 1999. Thus, applicant believes that the reference "Proposed SMPTE Standard for Television, SMPTE314M" was published in 1999 or earlier.

Claims 2 and 20 were objected to because of informalities. Claims 2 and 20 are currently amended to remove the informalities. Claims 15-16 were rejected under 35 U.S.C. 112, second paragraph, for insufficient antecedent basis for the limitation of "the string of bits". Claim 15 is amended to have sufficient antecedent basis.

Claims 1-48 were rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,397,324 (hereinafter "Barry"). Applicant respectfully disagrees.

Applicant respectfully submits that the pending claims are patentable over Barry.

For example, claims 4 and 5 recite:

> 4. (Original) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
>
> replacing at least one entry in at least one of a plurality of look-up units in a microprocessor unit with at least one number using a Direct Memory Access (DMA) controller;
>
> wherein the above operations are performed in response to the microprocessor receiving the single instruction.

5. (Original) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:

replacing at least one entry for each of a plurality of look-up units in a microprocessor with a plurality of numbers using a Direct Memory Access (DMA) controller;

wherein the above operations are performed in response to the microprocessor receiving the single instruction.

The Office Action relied upon the S2TBL instruction and the memory interface unit (485 of Fig. 4) of Barry for the rejection of claims 4 and 5.

The Office Action asserted that "... the memory interface unit (485 of Fig. 4), which is a DMA controller (see, Col. 7, lines 50-62)." Applicant respectfully submits that such an assertion is incorrect. A person skilled in the art understands that the memory interface unit 485 of Fig. 4 is not a DMA controller. There is no indication in Barry that the memory interface unit 485 of Fig. 4 is a DMA controller. In contrary, Fig. 1 of Barry shows a DMA controller 183 which is separate from the data memory interface 125.

Col. 7, lines 50-62, of Barry shows "the local data memories for the SP and each PE are currently organized as two memory banks to support independent, simultaneous accesses by the processing unit, and the direct memory access (DMA) controller". However, there is no indication in Barry about the type of access that involves a DMA controller.

Furthermore, there is no apparent relation between the DMA controller and the S2TBL instruction. Applicant respectfully submits that it is incorrect to assert that "the S2TBL instruction further specifies two pieces of data denoted by even and odd addresses (each pieces of data is considered a number) stored in the register file at Rte and Rto that will be written into the look-up table entries pointed to by the pointers created ...".

What stored in Rte and Rto of Barry are the two pieces of data, not the addresses of the two pieces of data. The two pieces of data stored in Rte and Rto are copied into the memory banks in response to the S2TBL instruction. See, for example, the "operation" column of the table in Fig. 8B. Thus, the S2TBL instruction loads data from the computer register file (CRF) into the memory banks (431 and 433), which would not involve a DMA controller; and the assertion in the Office Action is incorrect.

Further, claim 6 recites:

> 6. (Original) A method as in claim 5 wherein a single index encoded in the instruction specifies a location of the at least one entry in the plurality of look-up units.

In Barry, each of Rze and Rzo specifies one location in one memory bank; and Rze and Rzo are used to specify two locations in two memory banks. Clearly, this is different from using a single index to specify a location of **the** at least one entry *in the plurality of look-up units*. Note that claim 5 recites the limitation of "at least one entry for ***each*** of a plurality of look-up units".

Further, claim 7 recites:

> 7. (Original) A method as in claim 5 wherein a single index encoded in the instruction specifies a total number of the at least one entry for each of a plurality of look-up units.

In Barry, each of the STBL, S2TBL and S4TBL instructions is for one entry for each table, although different instructions may have different numbers of tables. There is no evidence that the STBL, S2TBL and S4TBL instructions have an index specifying a total number of entries ***for each*** table.

Further, claim 8 recites:

8. (Currently amended) A method as in claim 5 wherein a source address of the plurality of numbers in host memory is specified in an entry of a register file.

The S2TBL instruction of Barry uses Rte and Rto to store *the data* to be copied into the memory banks, *not the addresses*. The S2TBL instruction copies data from the computer register file (CRF) to the memory banks. The S2TBL instruction does not copy data from host memory into the memory banks. Thus, the S2TBL instruction does not contain a source address in host memory for the data to be copied.

Further, claim 10 recites:

10. (Original) A method as in claim 5 wherein an identity number encoded in the single instruction specifies the DMA controller.

Applicant respectfully submits that it is clearly incorrect to make the assertion "because the pointers are "encoded" in the instruction, and because all accesses to the look-up tables use the DMA controller (see Col. 7 lines 50-62), the instruction inherently specifies the DMA controller". The Office Action mistakenly considered the memory interface unit as a DMA controller. Further, in view of Fig.1 of Barry, Col. 7 lines 50-62, of Barry clearly indicates that the processing unit, or the direct memory access (DMA) controller, may access the memory banks. It is not necessary to access the memory banks (431 and 433) using a DMA controller. Furthermore, the logic of "not specifying a DMA controller" is inherently specifying a DMA is clearly improper.

Further, claim 49 recites:

49. (new) A method as in claim 5 wherein the at least one entry for each of the plurality of look-up units comprises a plurality of entries for each of the plurality of look-up units.

Barry does not show an instruction that loads *a plurality of entries* into ***each*** of a plurality of memory banks using a DMA. The S2TBL instruction of Barry loads only one entry into each of the two memory banks (431 and 433).

For example, claim 11 recites:

11. (Original) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
receiving a plurality of numbers;
partitioning look-up memory into a plurality of look-up tables;
looking up simultaneously a plurality of elements from the plurality of look-up tables, each of the plurality of elements being in one of the plurality of look-up tables and being pointed to by one of the plurality of numbers;
wherein the above operations are performed in response to the microprocessor receiving the single instruction.

The Office Action relied on Barry (Col. 7, lines 54-62 and Col. 9, lines 63-67) for the limitation of "partitioning look-up memory into a plurality of look-up tables". However, the description of Col. 7, lines 54-62 and Col. 9 lines 63-67 of Barry does not correspond to the claim limitation, since claim 11 recites the limitation "the above operations are performed in response to the microprocessor receiving the single instruction", which applies to "partitioning look-up memory into a plurality of look-up tables".

The Office Action relied on the L2TBL instruction of Barry for the rejection of claim 11. However, the description of Col. 7, lines 54-62 and Col. 9, lines 63-67 of Barry does not

show "partitioning" in response to the L2TBL instruction. Col. 7, lines 54-62, of Barry shows the hardware design choice of using "a multiple bank memory that makes it possible to generate multiple independent data-dependent load and store operations". Col. 9, lines 63-67, of Barry shows the design choices of using "both memory bank-1 431 and memory bank-0 433 simultaneously to support two load operations in parallel or two store operations in parallel". These design choices of using a memory with multiple banks cannot be considered corresponding to "*partitioning* look-up memory into a plurality of look-up tables, ... wherein the above operations are performed ***in response to*** *the microprocessor receiving the single instruction*".

Note that, in response to the L2TBL instruction, the processor of Barry uses the base addresses stored in the address register file (ARF) and offsets stored in the computer register file (CRF) to access the multiple bank memory. Since the look-up tables in the multiple bank memory of Barry are imaginary, based on how one interprets the meaning of the base addresses and offsets, the processor of Barry does not perform "partitioning look-up memory into a plurality of look-up tables, ... wherein the above operations are performed in response to the microprocessor receiving the single instruction".

Further, for example, claim 14 recites:

> 14. (Original) A method as in claim 11 wherein the look-up memory comprises a plurality of look-up units, and wherein said partitioning look-up memory comprises:
> configuring the plurality of look-up units into the plurality of look-up tables.

The description of Col. 7, lines 54-62 and Col. 9, lines 63-67 of Barry was relied upon in the Office Action for the limitation of "configuring the plurality of look-up units into the plurality of look-up tables". However, this portion of Barry is not about the operations that

are performed *in response to* the processor receiving L2TBL instruction. Note that claim 14 recites the limitation of "said partitioning look-up memory comprises: configuring the plurality of look-up units into the plurality of look-up tables" and claim 11 recites the limitation of "the above operations are performed in response to the microprocessor receiving the single instruction" which applies to "said partitioning".

Further, for example, claim 50 recites:

> 50. (new) A method as in claim 11 wherein the microprocessor comprises a media processor integrated with a memory controller for host memory on a single integrated circuit.

The memory interface unit (485) of a Barry is not a memory controller for host memory. The memory interface unit (485) is for local memory banks (431 and 433). Barry does not show "a media processor integrated with a memory controller for host memory on a single integrated circuit".

For example, claim 1 recites:

> 1. (currently amended) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
> receiving a first plurality of numbers and a second plurality of numbers, each of the first plurality of numbers pointing to one of a plurality of entries, each of the plurality of entries being in one of a plurality of look-up tables; and
> replacing simultaneously the plurality of entries in the plurality of look-up tables with the second plurality of numbers;
> wherein the above operations are performed in response to the microprocessor receiving the single instruction;

wherein the microprocessor is a media processor integrated with a memory controller for host memory on a single integrated circuit.

The memory interface unit (485) of a Barry is not a memory controller for host memory. The memory interface unit (485) is for local memory banks (431 and 433). Barry does not show "a media processor integrated with a memory controller for host memory on a single integrated circuit".

Other dependent claims and claims 25-48 directly recite the limitations discussed above or indirectly contain the above discussed limitations through dependency to the above discussed limitations. Thus, the pending claims are patentable over Barry.

Please charge any shortages or credit any overages to Deposit Account No. 02-2666. Furthermore, if an extension is required, Applicant hereby requests such extension.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 11/22, 2004

Lehua Wang
Reg. No. 48,023

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025-1026
(408) 720-8300